IN THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

Please cancel claims 26-31, 35-50 and 53.

1. (Previously Presented) A processor comprising:

a functional unit adapted to execute an instruction issued to it from a dispatch stage; and

a buffer in the dispatch stage coupled to the functional unit adapted to receive from a fetch stage and store a plurality of the instructions before issue to the functional unit and further adapted to store scheduling information associated with the instructions,

wherein the stored plurality of the instructions comprise a set of instructions from a loop body, and wherein the stored scheduling information defines, for each processor cycle in the loop, whether and which of the stored instructions are to be executed in that processor cycle, such that for a first processor execution cycle, the stored scheduling information is used to determine a first one of the stored instructions to be selected and issued to the functional unit from the buffer, and such that for a second different processor execution cycle, the stored scheduling information is used to determine a second one of the stored instructions to be selected and issued to the functional unit from the buffer.

2. (Original) A processor according to claim 1, further comprising:

a decode stage register coupled between the dispatch stage and the functional unit, the functional unit coupled to the decode stage register for executing the instruction issued to the decode stage register from the dispatch stage.

3. (Previously Presented) A processor according to claim 1, further comprising:

control logic coupled to the buffer adapted to cause a certain one of the stored plurality of instructions to be issued to the functional unit in accordance with a loop iteration stage and the stored scheduling information associated with the certain instruction.

4. (Previously Presented) A processor according to claim 3, wherein the control logic is further adapted to cause the certain one of the instructions to be issued in accordance with a cycle within the loop iteration stage and the stored scheduling information associated with the certain instruction.

5. (Previously Presented) A processor according to claim 1, wherein the scheduling information comprises a plurality of loop stage bit masks respectively associated with the plurality of instructions.

6. (Previously Presented) A processor according to claim 3,

wherein the scheduling information comprises a plurality of loop stage bit masks respectively associated with the plurality of instructions, and

wherein the control logic is further adapted to cause the certain one of the instructions to be issued in accordance with the respective one of the loop stage bit masks associated with the certain one of the instructions.

7. (Previously Presented) A processor according to claim 4,

wherein the scheduling information comprises a plurality of loop stage bit masks respectively associated with the plurality of instructions, and

wherein the control logic is further adapted to cause the certain one of the instructions to be issued in accordance with the respective one of the loop stage bit masks associated with the certain one of the instructions.

8. (Previously Presented) A processor according to claim 1, further comprising:

control logic coupled to the buffer, the control logic including:

an iteration initiation interval register for storing a loop iteration initiation parameter;

a loop iteration register for storing a loop iteration parameter; and

a loop cycles register for storing a loop cycles parameter,

wherein the control logic is adapted to cause the plurality of instructions to be issued to the functional unit from the buffer in accordance with the loop iteration initiation parameter, the loop iteration parameter, the loop cycles parameter and the stored scheduling information.

9. (Previously Presented) A processor according to claim 3, wherein the stored set of instructions consists of a kernel set of loop instructions, and wherein the control logic is operative so that the functional unit executes a number of loop iterations of the kernel set of loop instructions, a prologue set of loop instructions different than the stored kernel set of loop instructions, and an epilogue set of loop instructions different than the stored kernel set of loop instructions in accordance with the stored kernel set of loop instructions and received loop parameters.

10. (Previously presented) A processor according to claim 9, wherein the received loop parameters include a loop iteration initiation parameter, a loop iteration parameter and a loop cycles parameter, the control logic including:

an iteration initiation interval register for storing the loop iteration initiation parameter;

a loop iteration register for storing the loop iteration parameter; and

a loop cycles register for storing the loop cycles parameter.

11. (Previously Presented) A processor according to claim 1, wherein the set of instructions consists of a kernel set of loop instructions, the processor further comprising:

control logic coupled to the buffer, the control logic being operative so that the functional unit executes a number of loop iterations of the stored kernel set of loop instructions, a prologue set of loop instructions different than the stored kernel set of loop instructions, and an epilogue

set of loop instructions different than the stored kernel set of loop instructions based on the stored kernel set of loop instructions and received loop parameters.

12. (Previously presented) A processor according to claim 11, wherein the received loop parameters include a loop iteration initiation parameter, a loop iteration parameter and a loop cycles parameter, the control logic including:

an iteration initiation interval register for storing the loop iteration initiation parameter;

a loop iteration register for storing the loop iteration parameter; and

a loop cycles register for storing the loop cycles parameter.

13. (Previously Presented) A processor according to claim 2, wherein the set of instructions consists of a kernel set of loop instructions, the processor further comprising:

control logic coupled to the buffer, the control logic being operative so that the functional unit executes a number of loop iterations of the stored kernel set of loop instructions, a prologue set of loop instructions different than the stored kernel set of loop instructions, and an epilogue set of loop instructions different than the stored kernel set of loop instructions based on the stored kernel set of loop instructions and received loop parameters.

14. (Previously presented) A processor according to claim 13, wherein the received loop parameters include a loop iteration initiation parameter, a loop iteration parameter and a loop cycles parameter, the control logic including:

an iteration initiation interval register for storing the loop iteration initiation parameter;

a loop iteration register for storing the loop iteration parameter; and

a loop cycles register for storing the loop cycles parameter.

15. (Previously presented) A processor according to claim 3, further comprising a fetch unit, wherein the control logic is further operative so that the fetch unit can be shut down during execution a number of iterations of the loop body corresponding to the stored plurality of instructions.

16. (Previously presented) A processor according to claim 8, further comprising a fetch unit, wherein the control logic is further operative so that the fetch unit can be shut down during execution of a number of iterations of the loop body corresponding to the stored plurality of instructions.

17. (Original) A processor according to claim 11, further comprising a fetch unit, wherein the control logic is further operative so that the fetch unit can be shut down during execution of the prologue, kernel and epilogue sets of loop instructions.

18. (Original) A processor according to claim 13, further comprising a fetch unit, wherein the control logic is further operative so that the fetch unit can be shut down during execution of the prologue, kernel and epilogue sets of loop instructions.

19. (Previously presented) A processor according to claim 9, wherein the kernel set of loop instructions comprise modulo variable expansion (MVE) code.

20. (Previously presented) A processor according to claim 11, wherein the kernel set of loop instructions comprise modulo variable expansion(MVE) code.

21. (Previously presented) A processor according to claim 13, wherein the kernel set of loop instructions comprise modulo variable expansion (MVE) code.

22. (Original) A processor according to claim 3, wherein the control logic is further operative to allow interrupts to be handled at the end of a current loop iteration.

23. (Original) A processor according to claim 8, wherein the control logic is further operative to allow interrupts to be handled at the end of a current loop iteration.

24. (Original) A processor according to claim 11, wherein the control logic is further operative to allow interrupts to be handled at the end of a current loop iteration.

25. (Original) A processor according to claim 13, wherein the control logic is operative to allow interrupts to be handled at the end of a current loop iteration.

26-31. (**Canceled**)

32. (Previously Presented) A processor for executing a number of iterations of a loop, the loop including a prologue set of loop instructions, a kernel set of loop instructions and an epilogue set of loop instructions in accordance with a modulo schedule, the processor comprising:

a plurality of functional units; and

a dispatch stage coupled to the functional units for issuing instructions to the functional units, the dispatch stage including:

a plurality of buffers respectively associated with the plurality of functional units adapted to store the kernel set of loop instructions, and each buffer further adapted to store a different set of scheduling information associated with the respective functional unit and the stored instructions; and

control logic coupled to the plurality of buffers for causing the stored kernel set of instructions to be selectively issued to the respective functional units in accordance with the stored scheduling information, the control logic being operative so that the functional units execute the number of iterations of the kernel set of loop instructions, the prologue set of loop instructions and the epilogue set of loop instructions based on the stored kernel set of loop instructions and associated sets of scheduling information, wherein during a first processor cycle, the control logic uses the sets of scheduling information to cause a portion of the prologue set of loop instructions to be issued to the respective functional units from the buffers, and wherein during a second processor cycle after the first processor cycle, the control logic uses the sets of scheduling information to cause a portion of the kernel set of loop instructions to be issued to the respective functional units from the buffers, and wherein during a third processor cycle after the second processor cycle, the control logic uses the sets of scheduling information to cause a portion of the epilogue set of loop instructions to be issued to the respective functional units from the buffers.

33. (Original) A processor according to claim 32, further comprising a fetch unit, wherein the control logic is further operative so that the fetch unit can be shut down during execution of the number of iterations of the loop.

34. (Original) A processor according to claim 32, wherein the control logic is further operative to allow interrupts to be handled at the end of a current one of the number of loop iterations, and to complete the number of loop iterations after the interrupt is handled.

35-53. (**Canceled**)

54. (Previously Presented) A processor according to claim 1, wherein the buffer is further adapted to subsequently receive from the fetch stage and store a second different plurality of the instructions, thereby overwriting the previously stored plurality of instructions, before issue to the functional unit and further adapted to store second different scheduling information associated with the second plurality of instructions, wherein the second plurality of the instructions comprise a set of instructions from a second different loop body, and wherein the second instructions are issued to the functional unit from the buffer in accordance with the stored second scheduling information so as to cause the functional unit to execute a second number of iterations of the second loop body.